

The opinion in support of the decision being entered today (1) was not written for publication in a law journal and (2) is not binding precedent of the Board.

Paper No. 45

UNITED STATES PATENT AND TRADEMARK OFFICE

BEFORE THE BOARD OF PATENT APPEALS
AND INTERFERENCES

Ex parte BRYAN M. WILLMAN, DAN A. HINSLEY, JOHN D. VERT,
DAVID O. HOVEL and RITA M.C. WONG

Appeal No. 2000-1260
Application 08/518,852

ON BRIEF

Before FLEMING, LALL, and BLANKENSHIP, Administrative Patent Judges.

FLEMING, Administrative Patent Judge.

DECISION ON APPEAL

This is a decision on appeal from the final rejection of claims 23 through 29 and 34 through 40, all the claims pending in the application. Claims 1 through 22 and 30 through 33 have been cancelled.

The invention relates to a method and system for booting a computer. Specifically, the method and system provide for booting a computer system even after some configuration data has

become unusable. This is accomplished by booting the computer system from a set of configuration data that last booted the system properly. See Appellants' specification page 4 lines 19-24. The method and system include a BootSystem program (109) which is executed when a user request to boot the computer (105) is received via the input/output unit (125). See Appellants' Figures 1 and 2 and specification page 9 lines 21-24. The BootSystem program (109) invokes a ChoosesControlSet Program (113) which chooses a set of configuration data (*i.e.* a control set) with which to boot the computer (105). See Appellants' specification page 9 lines 26-30. The BootSystem program (109) then invokes an InitializeRegistry program (111) which, when executing on the CPU (127), retrieves configuration data for the computer (105) and stores the retrieved configuration data in the registry (107). See Appellants' specification page 9 lines 32-37. Next, the BootSystem program (109) invokes a LoadDeviceDrivers program (121) which, when executing on the CPU (127), attempts to load and initialize a set of device drivers listed in the chosen control set. See Appellants' specification page 9 line 37 - page 10 line 3. If any of the device drivers should fail to load or initialize, then the LoadDeviceDrivers program (121) decides whether the system boot procedure currently

underway should continue in spite of the failure to properly initialize the device driver, or whether a new system boot procedure should be initiated using the LastKnownGood control set (210) of the registry (107). See Appellants' specification page 10 lines 5-12. The LastKnownGood control set (210) contains a data value that identifies a control set containing the data values that last successfully booted the computer. See Appellants' specification page 8 lines 31-34. In response to a successful boot from the Default control set (209), the UpdateRegistry program (123) makes a new LastKnownGood control set (210) that is equivalent to the Default control set (209) which successfully booted the computer (105). See Appellants' specification page 10 lines 19-23. In this way, it is ensured that the computer (105) will successfully boot even after some configuration data in the Default control set (209) has become unusable. See Appellants' specification page 10 lines 30-33.

Independent claims 23 and 26 present in the application are reproduced as follows:

23. A method for configuring a computer system, the computer system having a plurality of devices and having a multiplicity of device drivers, each device driver for communicating with a device, the method comprising the computer-implemented steps of:

retrieving a first set of configuration data, the first set of configuration data having an association between devices and device drivers;

for each device driver associated with a device in the retrieved first set of configuration data,

loading the associated device driver; and

initializing the loaded device driver wherein the device driver returns an indication whether the device driver configured correctly;

automatically determining without receiving any user input whether each device driver associated with a device in the retrieved first set of configuration data initialized properly based on the returned indications; and

when it is determined that a device driver did not initialize properly,

retrieving a second set of configuration data, the second set of configuration data having an association between devices and device drivers such that, when the second set of configuration data was previously used when configuring the computer system, each device driver initialized properly; and

for each device driver associated with a device in the retrieved second set of configuration data,

loading the associated device driver; and

initializing the loaded device driver.

26. A method for booting a computer system, the computer system having a computer program and having a first and second set of configuration data, the first and second set of configuration data each specifying a configuration for the computer program, the second set of configuration data being a set of configuration data that was used when the computer program was previously booted successfully, the method comprising the computer-implemented steps of:

retrieving the first set of configuration data;

configuring the computer program in accordance with the
retrieved first set of configuration data;

automatically determining without receiving input from a
user whether the computer program was successfully configured in
accordance with the retrieved first set of configuration data;

when it is determined that the computer program was not
successfully configured in accordance with the retrieved first
set of configuration data, automatically performing the steps of:

retrieving the second set of configuration data; and

configuring the computer program in accordance with the
retrieved second set of configuration data; and

executing the configured computer program.

References

The references relied on by the Examiner are as follows:

Garner et al. (Garner)	5,014,193	May 7, 1991
Bertram et al. (Bertram)	5,261,104	Nov. 9, 1993
Archon (European Patent)	0 398 644	May 15, 1990

Rejections at Issue

Claims 23 through 25 and 34 through 36 stand rejected under
35 U.S.C. § 103 as being unpatentable over Archon, Bertram and
Garner. Claims 26 through 29 and 37 through 40 stand rejected
under 35 U.S.C. § 103 as being unpatentable over Bertram and
Garner.

Rather than repeat the arguments of Appellants or the Examiner, we make reference to the Briefs¹ and the Answer for the respective details thereof.

OPINION

After a careful review of the evidence before us, we do not agree with the Examiner that claims 23 through 25 and 34 through 36 are unpatentable under 35 U.S.C. § 103 over Archon, Bertram and Garner. Further, we do not agree with the Examiner that claims 26 through 29 and 37 through 40 are unpatentable under 35 U.S.C. § 103 over Bertram and Garner.

First we will address the rejection of claims 26 through 29 and 37 through 40 as being unpatentable under 35 U.S.C. § 103 over Bertram and Garner.

Appellants argue that the Bertram reference requires "manual indication to configure using the ROM-based default configuration data, whereas appellant's claims recite computer-implemented steps that avoid this manual indication." See page 15 lines 8-10

¹Appellants filed an appeal brief on November 17, 1997. In response to a Notification of Non-Compliance on February 18, 1998, Appellants files a second appeal brief (referred hereinafter as the Brief) on March 13, 1998. Appellants then filed a reply brief on August 17, 1998. The Examiner mailed an office communication on October 15, 1998 stating that the reply brief has been entered.

of the Brief. Appellants further argue that it is "[t]he Examiner's position that this difference would be obvious because it would 'allow booting of the system when the user does not know the specific manual indicia mentioned by Bertram cols. 7-8.' (Examiner's Action, October 16, 1996, p. 2.)" See page 15 lines 10-13 of the Brief.

Appellants then argue that "the Garner reference makes it particularly clear that it is concerned 'with dynamically configuring the presence or absence of an external storage device.' (Garner 3:43-45.)" See page 13 lines 12-14 of the Brief. Specifically, Appellants argue, with regards to column 3 lines 3-12 of Garner, that,

[t]his cited portion addresses determining peripheral configuration (*i.e.*, which devices are installed) and storing information related to changes in the configuration, which is performed without user interaction. The determining of peripheral configuration indicates which devices are currently installed so that the computer system can determine where it is located (*e.g.*, home or office). Appellants' claims, in contrast, are directed to 'configuring' a computer system (*e.g.*, an operating system), which is the actual process of modifying the system in accordance with configuration data.

See page 14 lines 9-16 of the Brief.

On page 2, lines 29-30 of the Answer, the Examiner sets forth the rejection of Appellants' claims 26 and 37 as being

unpatentable under 35 U.S.C. § 103 over Bertram and Garner.

Then, on page 5 lines 1-5 of the Answer, the Examiner states that claims 26 and 37 do not recite the device drivers claimed in claim 23 and therefore the rejection is the same as the rejection for claim 23 (see page 2 lines 26-28 of the Answer) with the exception that the Archon reference is no longer relied upon.

In specific reference to the subject matter of the rejection of claim 26, the Examiner states that Bertram shows that upon a determination of an improper initialization with regards to a first set of configuration data, "retrieving a second set of configuration data, such that, when the second set of configuration data was previously used when configuring the computer system," there was proper initialization based on the default configuration, column 8 lines 31-32. See page 3 lines 10-17 of the Answer. The Examiner then states that "Garner et al. explicitly teaches performing initial configuration of the system without user input (col. 3 lines 3-12)." See page 3 lines 21-23 of the Answer. The Examiner further states that "Garner further teaches storing the last configuration so that any changes may be detected and the configuration may be updated (col. 2 lines 7-9)." See page 4 lines 15-17 of the Answer.

In rejecting claims under 35 U.S.C. § 103, the Examiner bears the initial burden of establishing a **prima facie** case of obviousness. *In re Oetiker*, 977 F.2d 1443, 1445, 24 USPQ2d 1443, 1444 (Fed. Cir. 1992). Further, our reviewing court in *In re Dembiczak*, 175 F.3d 994, 999-00, 50 USPQ2d 1614, 1617 (Fed. Cir. 1999) has said,

Broad conclusory statements regarding the teaching of multiple references, standing alone, are not 'evidence.' *E.g.*, *McElmurry v. Arkansas Power & Light Co.*, 995 F.2d 1576, 1578, 27 USPQ2d 1129, 1131 (Fed. Cir. 1993) ("Mere denials and conclusory statements, however, are not sufficient to establish a genuine issue of material fact."); *In re Sichert*, 566 F.2d 1154, 1164, 196 USPQ 209, 217 (CCPA 1977).

We note that Appellant's claim 26 recites the following:

. . . the second set of configuration data being a set of configuration data that was used when the computer program was **previously booted successfully**. . . ,

automatically determining without receiving input from a user whether the computer program was successfully configured in accordance with the retrieved first set of configuration data;

when it is determined that the computer program was not successfully configured in accordance with the retrieved first set of configuration data, **automatically performing** the steps of:

retrieving the second set of configuration data . . .

(Emphasis added).

In looking to Appellants' specification for support for the above limitations, we find that the critical limitations are the steps of "**automatically determining without receiving input from a user** whether the computer program was successfully configured in accordance with the retrieved first set of configuration data, determining that the computer program was not successfully configured and automatically retrieving the second set of configuration data." As stated *supra*, we find that Appellants disclose the steps and program for "automatically determining" and "automatically retrieving" on page 10 lines 5-17 and in Figures 1, 2 and the last paragraph of computer program in Figure 5,

[i]f any of the device drivers should fail to load or initialize, then the **LoadDeviceDrivers program (121)** **decides** whether the system boot procedure currently underway should continue in spite of the failure to properly initialize the device driver, or whether a new system boot procedure should be initiated using the LastKnownGood control set (210) of the registry (107). **If a decision is made to reboot the computer (105) using the LastKnownGood control set (210) then the system boot procedure is reinitiated using the configuration data stored in the LastKnownGood control set (201).**

(Emphasis added). We find further disclosure supporting Appellants claim limitations in a flow chart illustrated in Figure 7 and defined on page 14 lines 5-9,

if the user did not hit the breakin key then in step (717) **the method ChooseControlSet determines** whether the LastKnownGood environment variable is set equal to the value True. If the LastKnownGood environment variable is set equal to the value False, then processing continues

(Emphasis added).

Upon careful review of Bertram, we find that the user "must" reboot the system in order to correct the failed startup. Specifically, we find that Bertram teaches that a user can reset the computing system to a default configuration,

if he [the user] has written a bad DOS start up file or has deleted the partition on his hard disk and doesn't have a bootable diskette, he can recover and still be able to use the computing system. The technique calls for the initializing routine to check manually operated indicia for a **user initiated** condition during the power up sequence. This condition can be the holding down of the mouse button or holding a specified key or set of keys . . . when the power switch is thrown. During power up, the initializing routine immediately checks for the state of the **manually operated indicia** . . . [and if] it is in the designated state . . . the **initializing routine then resets** the bits in the customizing word in RAM which defines the system start up options. **By resetting the options to their default state,** the machine is thereby returned to a known state.

(Emphasis added). See column 7 line 60 through column 8 line 32 of Bertram. Though we note our finding supports the Examiner's position that the Bertram reference is not automated, we fail to find that Bertram suggests the retrieval of a second set of

configuration data being a set of configuration data that was used when the computer program was previously booted successfully.

Upon careful review of Garner, we find that Garner's invention is motivated by overcoming past problems that occurred when a user moved an "already bootable computer" from one place to another and the peripherals changed. See column 1 lines 35-50 of Garner. Specifically, we find that Garner is **not** directed to the problems of "initial booting" of a computer but rather to the problems of being able to switch between one peripheral and another "after the system is booted." See column 1 lines 58-64. This is done by the accessing of a database that contains information for the different peripherals that might be used (*i.e.* a printer at a home office or a different type of printer at a work office) and loading the appropriate data to allow interaction between said "already" booted computer with the peripheral of choice. See column 2 lines 12-17 and claim 1 of Garner. To further support the finding above, we note that Garner specifically discloses a system that provides for the dynamic configuration system of a portable computer that contains,

additional peripheral status or configuration storage registers and a novel software driven means for dynamically reconfiguring the system without user interaction. The system of the present invention includes means for determining the peripheral status and updating the configuration data to permit use of the computer to begin without executing a user prompted initialization sequence or requiring manual switches or jumpers to be reset each time the user changes environments.

See column 1 lines 53-64 of Garner.

Upon reviewing both Bertram and Garner *supra*, we find nothing in either reference that supports the Examiner's position that either of these references teach the limitations of claim 26 as previously discussed.

In addition, Appellants argue that "[t]he Combination of the Bertram and Garner References Does Not Suggest Retrieving a Second Set of Configuration Data in Response to an Automated Determination That a First Set Of Configuration Data Was Not Successfully Used For Configuration". See page 14 lines 24-27 of the Brief. The Appellants further argue that the combination of Bertram and Garner "would still use **Bertram's manual indication to configure using the ROM-based default configuration data, whereas appellants' claims recite computer-implemented steps that avoid this manual indication.**" (Emphasis added). See page 15 lines 8-10 of the Brief. Appellants also argue that "[t]he

Appeal No. 2000-1260
Application 08/518,852

Examiner has provided absolutely no support as to why appellants' solution would be an obvious choice." See page 15 lines 23-24 of the Brief.

In providing motivation or a suggestion to combine, we find that our reviewing court states in *In re Lee*, 277 F.3d 1338, 1342-43, 61 USPQ2d 1430, 1433 (Fed. Cir. 2002),

[t]he essential factual evidence on the issue of obviousness is set forth in *Graham v. John Deere Co.*, 383 U.S. 1, 17-18, 148 USPQ 459, 467 (1966) and extensive ensuing precedent. The patent examination process centers on prior art and the analysis thereof. When patentability turns on the question of obviousness, the search for and analysis of the prior art includes evidence relevant to the finding of whether there is a teaching, motivation, or suggestion to select and combine the references relied on as evidence of obviousness. See, e.g., *McGinley v. Franklin Sports, Inc.*, 262 F.3d 1339, 1351-52, 60 USPQ2d 1001, 1008 (Fed. Cir. 2001) ("the central question is whether there is reason to combine [the] references," a question of fact drawing on the Graham factors).

We find that the Examiner states on page 4 lines 18-25 of the Answer that,

it would have been obvious to one of ordinary skill in the art to use a default configuration as taught by Bertram et al so that a working configuration of the system can be obtained (Bertram et al col. 7 lines 45-66) and automatically determine . . . [proper initialization] in order to allow booting of the system when the user does not know the specific manual indicia mentioned by Bertram [sic] on cols. 7-8.

Having reviewed the Bertram and Garner references, we find no factual basis or motivation for suggesting their combination as suggested by the Examiner. Further, we find no evidence to modify the Bertram reference, in light of Garner, to automate the boot sequence as suggested by the Examiner. Rather, we find, as stated *supra*, that the Garner invention is concerned only with a "post-boot-up" problem and fails to suggest any reason for addressing a "computer booting" problem as relates to Bertram and Appellants' claimed limitations.

Therefore, we will not sustain the Examiner's rejection of claims 26 through 29 and 37 through 40. Furthermore, we note that the Examiner relies on Bertram and Garner for the above discussed limitations in rejected claims 23 through 25 and 34 through 36. Therefore, we will not sustain the rejection of these claims.

Appeal No. 2000-1260
Application 08/518,852

In view of the foregoing, the decision of the Examiner
rejecting claims 23 through 29 and 34 through 40 under 35 U.S.C.
§ 103 is reversed.

REVERSED

MICHAEL R. FLEMING)	
Administrative Patent Judge)	
)	
)	
)	BOARD OF PATENT
PARSHOTAM S. LALL)	
Administrative Patent Judge)	APPEALS AND
)	
)	INTERFERENCES
)	
HOWARD B. BLANKENSHIP)	
Administrative Patent Judge)	

MRF:pgg

Appeal No. 2000-1260
Application 08/518,852

John B. Conklin
Leydig Voit & Mayer, LTD
2 Prudential Plaza
Suite 4900
Chicago, IL 60601-6780